L17: Entry 1 of 1 File: TDBD May 1, 1995

TDB-ACC-NO: NN9505419

DISCLOSURE TITLE: Reducing the Latency of Distributed Resource Registration

PUBLICATION-DATA:
IBM Technical Disclosure Bulletin, May 1995, US

VOLUME NUMBER: 38
ISSUE NUMBER: 5
PAGE NUMBER: 419 - 420

PUBLICATION-DATE: May 1, 1995 (19950501)

CROSS REFERENCE: 0018-8689-38-5-419

DISCLOSURE TEXT:

This is a technique for reducing the amount time it takes for a "Resource" to be registered (by a registrant) for involvement in a distributed transaction that is managed by an implementation of the Object Transaction Service (OTS). The behavior of the OTS is specified by the Object Management Group (OMG) as an Object Service component of the Common Object Request Broker Architecture (CORBA). The reduction in time is only apparent in cases where the transaction is distributed and is observed by the registrant when its local "Coordinator" is a subordinate. The technique can improve throughput and reduce the occurrence of time-outs: the advantage of its effect will increase as the distance in the path between the registrant's "Coordinator" and the root "Coordinator" object increases in the "commit digraph." According to the OTS specification (OMG TC document 94-8-4), a "Resource" is not involved in a transaction until it is registered with its local "Coordinator" object. In turn, a "Coordinator" is not involved in any of its superiors' transactions until a local "Resource" is registered for involvement in one of those transactions. A set of transactional operations can be invoked on "TransactionalObject"s in a single transaction; objects in this set are related in a "request digraph." Any path in this graph can span the domains of several "Coordinator"s. Until the first "Resource" is registered for involvement in the transaction there is no corresponding commit digraph. The problem is most easily recognized when a "Resource" at the end of a long "request path" is first registered and a corresponding "commit path" must be constructed. The commit path is constructed as a side effect of the "Resource"'s "Coordinator" registering itself with its superior and each superior "Coordinator" recursively registering itself with its own superior.

If the commit path is constructed by "Coordinator"s registering themselves by invoking normal synchronous "Coordinator::register_resource" operations on their superiors, then the time between the first "Resource" invoking "register_resource" and receiving the corresponding reply could easily exceed a local timeout interval, especially if a long series of inter-machine and inter-process communications need to be carried out. If, as with the method we employ, the commit path is constructed as the side effect of "Coordinator"s asynchronously registering with their superiors, then the reply to the initiating registrant can then be made immediately, without having to wait for the requests and replies to all the "Coordinator"s in the newly constructed commit path.

The OTS is being built on top of the IBM* System Object Model (SOM) and its extension for distribution: (DSOM) Distributed SOM. The invention requires the use of asynchronous operation invocations - or as they are referred to in the OMG's Common Object Request Broker Architecture (CORBA) deferred synchronous operations. The implementation of these are provided in DSOM by the CORBA-compliant "Request::send" and "Request::get_response" operations.

A cost effect of the invention is that the recovery logic of the "Coordinator" design is made more complex than in the case with synchronous registration of a subordinate "oordinator" with a superior; the "Coordinator" has to account for the fact that communications somewhere in the path to the root "Coordinator" may fail before the commit path has been constructed fully. The originating "Coordinator" must guard against the effects of incorrectly informing a registrant of the successful outcome of its "register_resource" operation before the commit path has been constructed in such a way that it can be recovered. This is a solution to a new problem that results from the combination of objects and transactions. Static registration of Resource Managers (RMs) with Transaction Managers (TMs) is an assumption that underlies existing TMs and RMs even though dynamic registration is specified in the X/Open XA architecture. In contrast, the OMG Object Transaction Service only specifies dynamic registration of "Resource" objects with "Coordinator"s. More importantly, the advent of object technology permits an increased granularity of "Resource" objects and, together with the rapid increase in distributed computing, has resulted in a far greater potential for the problem we are addressing to occur and thus for the solution we employ to have value. * Trademark of IBM Corp.

Previous Doc Next Doc Go to Doc#